COMPILADORES
Proceso de compilación

SUAYED
Ricardo Mancilla Guzmán
410110724
ricardomancillag@hotmail.com
rickman@comunidad.unam.mx

Resumen

U compilador funciona como una secuencia de fases cada una de las cuales transforma una representación del programa fuente en otro,

Introducción

En la práctica las fases pueden agruparse en análisis y síntesis, pero si nos adentramos más en el tema podemos estructurar el proceso de compilación en 5 grandes etapas.

Título Principal

PROCESO DE COMPILACION

En la práctica las fases pueden agruparse en análisis y síntesis, pero si nos adentramos mas en el tema podemos estructurar el proceso de compilación en 5 grandes etapas.

1. Analizador de léxico

Realiza otras tareas aparte de identificar lexemas, entre estas se encuentran: eliminar los cometarios y espacios en blanco, correlacionar con e programa fuente los mensajes de error generados por el compilador.

2. Analizador sintáctico

La sintaxis de construcción de un lenguaje de programación puede especificarse mediante gramáticas libres de contexto o mediante connotación BNF.

3. Analizador semántico

Es la tercera fase del compilador entre sus funciones se encuentran: utiliza árbol sintáctico y la información en la tabla de símbolos para comprobar la consistencia semántica del programa, recopilar información sobre el tipo de árbol sintáctico, verificar que cada operador tenga operando que coincidan y reducir un código intermedio

4. Generador de código intermedio

Consiste en que el compilador verifique línea por línea todas y cada una de las expresiones escritas dentro del editor del lenguaje, verificando en las librerías la existencia de cada palabra reservada, tipo de dato.

5. Optimizador de código

Esta fase es independiente del proceso y tiene como objetivo mejorar el código intermedio, de manera que se produzca un mejor código destino, mejorar en términos de tiempo, de requerimiento de memoria y de uso de recursos.

Durante el análisis se divide el programa fuente en componente y se genera una estructura gramatical,

después se utiliza para crear una representación intermedia del programa fuente.

Si durante el análisis se detecta que el programa fue esta mal al respecto a la sintaxis o semántica se genera un mensaje para que el programador los corrija, también se recolecta información sobre el programa fuente que es almacenado en una estructura denominada tabla de símbolos, la cual se pasa junto con la representación intermedia a la parte de síntesis y es utilizada durante toda las fases de la compilación.

En la parte de síntesis se construye un programa objeto.
El análisis se le conoce como front end del compilador a la síntesis se la traducción propiamente dicha, se le denomina back end.

Analisis léxico grafico